

複数タブレット画面による仮想ディスプレイのための同期的通信手法

佐々田 惟可[†] 大野 佑樹[†] 清水 祐弥^{††} 高崎 尚人^{††} 小林 亜樹[†]

[†]工学院大学工学部情報通信工学科 ^{††}工学院大学大学院工学研究科電気・電子工学専攻

1 はじめに

グループミーティングではテーブルトップ PC 環境での作業が有用である場合がある。しかし、テーブルトップ PC は一般的に普及していない。そこで、近年幅広く普及しているタブレット端末を用いて、テーブルトップ PC と同様の環境を構築する手法が提案されている [1]。また、仮想テーブルトップ環境におけるコラボレーション効果に関する研究 [2] など進められている。これらは同仮想環境の可搬性を重視し、タブレット端末等のうち 1 台をホストとして利用する構成となっている。

しかし、画面の多画面化や構成端末数の増加に伴い全体を統括するホストに処理が集中し、ad-hoc な通信モードと併せてスケーラビリティ上の問題となる可能性がある。

そこで本研究では、インターネット越しに十分な性能を持ったホストコンピュータによる統括制御を指向するシステム構成を提案する。本論文ではこのうち、ネットワーク遅延の影響により接続端末間において描画の同期性が失われる現象などを抑制し、良好なテーブルトップ環境を構築するための画面情報通信方式について提案する。

2 仮想テーブルトップシステム

本研究では、単一の仮想テーブルトップ環境を統括管理する主体として十分な処理能力をもったコンピュータを想定する。このホストは、インターネット越しに HTTP (あるいは HTTPS) アクセスできる環境とすることで利用環境を選ばないようにする。

システムの全体像は図 1 のとおりで、画面全体、端末認証、識別などのシステムを左側のサーバ上で管理する。認証を済ませた各端末はクライアント主導による HTTP 通信で必要な情報を得る。必要な画面矩形の配信を受け、入力情報をサーバへ送信する。

このような非対称な構成とすることで、特定のタブレット端末にテーブルトップ環境全体の責任を負わせ

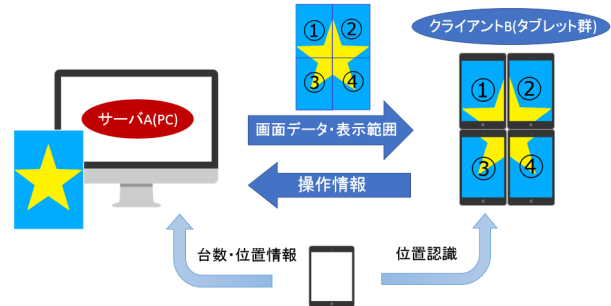


図 1: システムイメージ

ることなく、データ損失などを防ぎ、安定的な運用を行えるようにする。将来的には各端末を入力デバイスとして外部リソースの取り込みを行った結果をサーバ上で保管する機能と組み合わせて、コラボレーション効果を増すことも視野に入れた設計である。

本論文ではこのうち、画面情報の配信手法について提案を行う。画面描画ではネットワーク遅延が問題となり、また、各端末への配信が同時とはならないためのずれが発生するなどの問題がある。本手法はこれらの問題発生を抑制する同期的配信手法である。

3 画面情報通信方式

TCP 接続を維持した状態（キープアライブ）で HTTP 通信によりデータを配信する。配信するデータは以下の 4 種類である。

- 画面全体のデータ
- 表示範囲座標値
- ハートビート
- 無効矩形

まず、クライアントからのリクエストにより初回の通信で画面全体のデータを配信する。はじめに各端末に画面全体のデータを配信することでユーザ操作以降の通信は変更された表示範囲の座標値のみとなる。その後サーバから全クライアントに対して一定の間隔でハートビートを配信し、通信の安定性を確認しておく。

ユーザが端末を操作すると操作した画面の座標情報をサーバに転送し、転送された座標情報をもとに各端末の表示範囲座標を変更し各クライアントに配信する。これによりユーザが操作していない端末に対しても表示範囲の更新が行えるため、複数の端末画面を同期的に動作させることができる。

Synchronous communication method for a virtual display by multi-tablets

Yuika Sasada[†] Yuki Ono[†] Yuya Shimizu^{††}
Naoto Takasaki^{††} Aki Kobayashi[†]

[†] Department of Information and Communications Engineering, Faculty of Engineering, Kogakuin University

^{††} Electrical Engineering and Electronics, Kogakuin University Graduate School

仮想テーブルトップ画面の変更が生じた際には、変更箇所を含む無効矩形領域が通知される。各クライアントは、当該領域の画面情報を逐次同様の手順でサーバへ要求し、端末内に保管しておく。

4 システム構成

各端末のクライアントアプリケーションの動き及びサーバクライアント間の HTTP 通信について、画面描画処理と同期的画面動作に分けて説明する。

4.1 画面描画処理

クライアントアプリケーションは図2のように以下の手順で画面描画を行う。

1. サーバに対して HTTP GET を送る。
2. 画面全体のデータを取得する。
3. decodeStream メソッドで取得した画面全体のデータを Bitmap に変換する。
4. Bitmap データの縦横サイズを getHeight, getWidth メソッドで取得する。
5. それぞれの端末が表示すべき描画範囲の座標値を計算する。
6. Bitmap データを編集可能な Canvas 型に変換する。
7. 描画範囲座標値の部分を取り取る。
8. setImageBitmap メソッドで表示する。

サーバから画面データを取得する際に、あらかじめ分割せず画面全体のデータを取得することで画面の配信は一度で済み、ユーザによる操作後の二度目以降の通信は画像の描画範囲の座標値のみになるためデータ量が小さくなり、それに伴い配信に係る遅延のばらつきを抑えることができる。

4.2 同期的画面配信

ユーザが手近な1台のタブレット端末をスワイプ操作した際の HTTP 通信について図3を用いて説明する。画面描画の手順については4.1節で説明したのでここでは除く。ユーザが操作する端末をクライアント1、操作しない他の端末をクライアント2とする。サーバはクライアント1, 2に対してハートビートを配信し TCP 接続を維持しているものとする。

1. ユーザがクライアント1の画面に指で触れ、画面をスワイプする。
2. 端末画面に指で触れた座標及び指を移動し画面から離れた座標を取得する。
3. 取得した座標情報をサーバに送信する。
4. サーバ上で各端末の描画範囲座標値を計算し直す。
5. 修正した座標値を各端末に配信する。
6. 配信された座標に対応する画面に更新する。

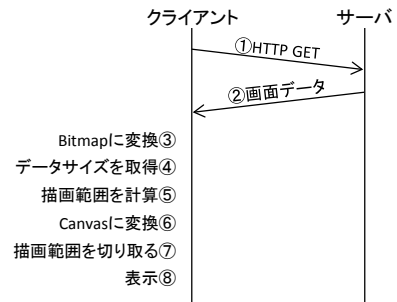


図2: 画面描画処理

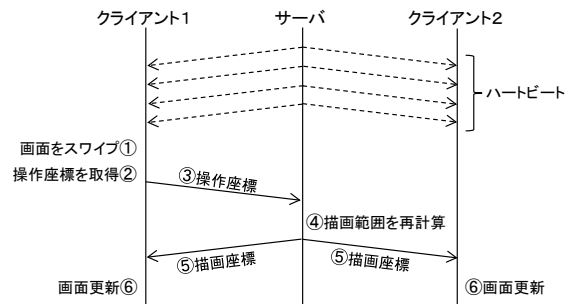


図3: 同期的画面配信

サーバからクライアントに対してハートビートを配信し通信の安定性を確認しつつ、操作していない端末に対してサーバから表示範囲座標値を配信することができる。これにより1台の端末の操作情報を全端末に反映させ、ユーザのタップ操作に対して複数の端末画面を同期的に動作させることができる。

5 おわりに

複数のタブレット端末を用いた同期的な画面情報通信方式を提案した。

本手法は、初回の通信で画面全体のデータを配信し、二度目以降の通信を描画範囲の座標値のみで通信量を抑え、接続端末ごとに描画に必要な情報の到達時刻に差異が生じ、画面描画の同期性が失われる現象などを抑制する。

今後は試作システムを用いて各端末の画面描画時のずれについて有用性の評価を行う。

参考文献

- [1] Arne Schmitz, et al. “Ad-Hoc Multi-Displays for Mobile Interactive Applications”, 31st Annual Conference of the European Association for Computer Graphics, Vol.29, No.2, pp.45-52(2010)
- [2] 伊藤直人, 高田秀志 “タブレット端末を用いた仮想テーブルトップ環境の協調作業への適用とその評価”, 第93回GN研究会, Vol.2015-GN-93, No.17, pp.1-6(2015)